# An Access Etiquette for Very-Wide Wireless Bands*

Rodrigo Garcés , J.J. Garcia-Luna-Aceves
Computer Engineering Department
School of Engineering
University of California at Santa Cruz
Santa Cruz, CA 95064, USA
garces/jj@cse.ucsc.edu

Raphael Rom
Sun Microsystems Laboratories
901 San Antonio Road
Palo Alto, CA 94303, USA
raphael.rom@eng.sun.com
and
Department of Electrical Engineering
Technion, Israel

## Abstract

*We propose and analyze a specific set of access rules, or "spectrum etiquette," for the 59-64 GHz unlicensed band to allow systems from different manufacturers with different physical and medium-access control protocols to co-exist, sharing the large available bandwidth without interference. The proposed etiquette is unique in that heterogeneous systems are able to co-exist with one another, without monitoring the entire band, by means of transmissions over a common, narrow band control channel used to establish collision-free transmission schedules over the channels allocated for data transmission within the 59-64 GHz band. Because no common physical layer can be assumed among different systems, the control channel is needed for the systems to schedule transmissions in the rest of the band, and the only means by which systems can communicate with one another over the control channel is the duration of each others' transmissions, which are perceived only as noise. A transmission encoding is defined based on this basic feedback to allow systems to ascertain which system can use which data channel at which time without interference. Analytical and simulation results are presented showing that the proposed etiquette is fair to all the co-existing systems, fully utilizes the spectrum, provides bounded delays for data-channel acquisition time by any given system, and provides minimum channel-use guarantees.*

## 1 Introduction

In the U.S., the Federal Communications Commission (FCC) made available 6.2 GHz of spectrum and established technical rules that permit the introduction and development of communications technologies in the millimeter wave frequency bands above 40 GHz [4]. Europe and Japan are also considering commercial uses of millimeter wave technology.

The term "millimeter wave" is taken from the fact that the wavelength of radio signals between 30 GHz and 300 GHz ranges from 10 millimeters down to 1 millimeter. The FCC action makes available three frequency bands: 47.6-47.8 GHz, 59-64 GHz, and 76-77 GHz, for unlicensed vehicle radar systems and general purpose unlicensed devices. The 59-64 GHz band was set aside as a general unlicensed band. This is an unprecedented decision in terms of bandwidth being made available and the lack of regulatory constraints.

The 59-64 GHz band could be used for wide bandwidth computer communication over point-to-point wireless links at data speeds that may exceed 5 Gbps. This would extend the data rates currently available to a fixed user through fiber optic cable. However, equipment may not be operated on this band, until an etiquette has been defined for its use. In this context, an etiquette is a specific set of access rules that permits multiple systems from different manufacturers to share the bandwidth without undue interference, and without requiring the manufacturers to adhere to the same medium-access control (MAC) or physical-layer (PHY) communication protocols. Several companies, including Hewlett-Packard, Apple, Sun, Motorola, Hughes Research, Eaton Division of Cutler-Hammer, Rockwell International, and Metricom among others, have begun defining a spectrum etiquette for sharing this band [6]. The main challenge in the definition of such an etiquette is the huge bandwidth that is being made available, which precludes systems from listening over the entire band to try to prevent interference. According to the FCC regulations [4], within the 59-64 GHz band, the power density of any emission shall not exceed $9\mu W/cm^2$ at a distance of 3 meters. The power density of any emissions outside the 59-64 GHz band must consist solely of spurious emissions and must not exceed $90pW/cm^2$ at a distance of 3 meters. The power measurements will be average measurements based on a 1 MHz bandwidth. Within this constraints, the etiquette should fulfill the following requirements [4, 6]:

- The etiquette should provide a substantial reduction in the probability of interference between co-existing systems.
- The etiquette should seek to promote realization of high-speed communications while attempting not to foreclose low speed communications.
- The etiquette should be flexible enough to allow as many applications as possible to effectively co-exist in the band.
- The etiquette should not have a major negative impact on the economic feasibility of systems.
- The etiquette should provide for the diverse needs of both continuous-connection and burst-mode systems.
- In all portions of the band where etiquette applies, only one etiquette should be used.
- The etiquette must be kept simple. To this end, effectiveness may be traded off for simplicity. The etiquette must use as few layers as possible in the standard OSI stack.
- The etiquette must promote efficient use of the spectrum.
- The etiquette must be open and non-proprietary, it must have openly-available set of procedures.

In this paper, we propose a listen-before-transmit etiquette for heterogeneous systems implementing different PHY and MAC protocols and based on power sensing over a control channel used to schedule access to the rest of the band, which is partitioned into data channels. Each system consists of any set of nodes using the same PHY and MAC protocols, and two nodes from different systems cannot decode one another's transmissions in any data channel of the band. Each data channel is meant to be used by an individual system (i.e., two or more nodes using the same PHY and

*The work at UCSC was supported in part by the Defense Advanced Research Projects Agency (DARPA) under grant DAAB07-95-C-D157 and by UC MICRO grant with SUN Microsystems Laboratory.

# Report Documentation Page

*Form Approved*
*OMB No. 0704-0188*



| | | |
|---|---|---|
| 1. REPORT DATE<br>**1998** | 2. REPORT TYPE | 3. DATES COVERED<br>**00-00-1998 to 00-00-1998** |
| 4. TITLE AND SUBTITLE<br>**An Access Etiquette for Very-Wide Wireless Bands** | | 5a. CONTRACT NUMBER |
| | | 5b. GRANT NUMBER |
| | | 5c. PROGRAM ELEMENT NUMBER |
| 6. AUTHOR(S) | | 5d. PROJECT NUMBER |
| | | 5e. TASK NUMBER |
| | | 5f. WORK UNIT NUMBER |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br>**University of California at Santa Cruz,Department of Computer Engineering,Santa Cruz,CA,95064** | | 8. PERFORMING ORGANIZATION REPORT NUMBER |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) | | 10. SPONSOR/MONITOR'S ACRONYM(S) |
| | | 11. SPONSOR/MONITOR'S REPORT NUMBER(S) |
| 12. DISTRIBUTION/AVAILABILITY STATEMENT<br>**Approved for public release; distribution unlimited** | | |
| 13. SUPPLEMENTARY NOTES | | |
| 14. ABSTRACT | | |
| 15. SUBJECT TERMS | | |

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON |
|---|---|---|---|---|---|
| a. REPORT<br>**unclassified** | b. ABSTRACT<br>**unclassified** | c. THIS PAGE<br>**unclassified** | | **8** | |

MAC layers) on long-term and persistent basis. The control channel is used to exchange information about the band use activity in the area. Prospective transmitters listen to the control channel to get information about the data channels occupancy; this eliminates the need to listen to the entire wide band.

The only means by which systems can exchange information with one another over the control channel is the duration of each others' transmissions, which are perceived only as noise, and no information is exchanged across systems over the data channels defined in the band. A novel transmission encoding is defined based on this basic control-channel feedback that allows systems to ascertain which system can use which data channel at which time, without interference.

The proposed etiquette is meant for assignment of data channels to systems, rather than individual stations. Hence, it makes sense to have the data channels be of substantial width such as, for example, to support OC3 rates. For the 59-64 band, this means that the number of data channels is around 30.

The remainder of the paper is organized as follows: Section 2 describes the assumptions that we make for our model. Section 3 gives a detailed description of the control channel and the proposed etiquette basic operation. Section 4 presents analytical and simulation results showing that the proposed etiquette is fair for all the co-existing systems, fully utilizes the spectrum, provides bounded delays for data-channel acquisition time by any given system, and provides minimum channel-use guarantees. The results show that the etiquette's use of the available band approximates that of an optimal assignment of data channels to co-existing systems. Section 5 offers our concluding remarks.

## 2 Definitions and Assumptions

Throughout this paper, the *band* for which the proposed etiquette is used is the 59-64 GHz general unlicensed band. In our model, the band is divided into $s$ *data channels* (i.e., channels that are used to transmit data once the channel has been acquired) and a *control channel* (i.e., a small predefined portion of the band for the exchange of scheduling information among the systems). In our model, a *system* is a collection of nodes sharing the same PHY- and MAC-layer protocols. We make no assumptions on the way in which a system allocates the data channels to its nodes or schedules their use. However, we adopt a system-level approach in the operation of the etiquette. More specifically, we assume that a single station in a system is in charge of participating in the etiquette activity in the control channel to obtain usage rights for data channels. We refer to this station as the *etiquette designated station (EDS)*. We note that the EDS for a system need not always be the same i.e., a system can change its EDS at will. The idea of an EDS is to ensure that every system is represented only once, so that stations from the same system do not compete in the control channel. The EDS is also in charge of notifying the rest of the stations in its own system about availability of data channels. The EDS is the only node in the system that is allowed to send reservation signals in the control channel, all other nodes in the system are only allowed to send an *echo*, i.e., a response to such a reservation signal.

We assume that no information is exchanged among the systems over any of the data channels. Each system is fully independent in that its PHY- and MAC-layer protocols which can be completely different from any other system's PHY/MAC layer.

A unique identifier (ID) is assigned to each EDS, which in practice could consist of three fields: the device's FCC ID number; the device's serial number; and a user-definable field.

The control channel is organized in *frames*, each of which is further divided into *periods* made up of several slots; the exact structure of these is further discussed below. The number $s$ of data channels is assumed to be predefined, and that number determines the length of a frame in the control channel.

## 3 Etiquette Description

In general, the etiquette described in this paper can be defined as a specific set of access rules that permits multiple systems with different PHY and MAC protocols to compete for channel usage one channel at a time on the 59-64 GHz band.

A system cannot compete for a specific channel but rather for whichever channel happens to be available next. A system cannot target a specific channel but must use whichever channel it happens to acquire. The allocation of channels is done in order and a system cannot compete for a second data channel, until all other systems acquire their first data channel. A system may not be removed from its current data channel if the number of systems is smaller than the number of data channels. If there are more systems than data channels, then the system may be removed from the data channel it has acquired.

Competition for a data channel is resolved by way of a collision-resolution algorithm. This algorithm requires that the activity on a channel (idle, success, collision) be known to the sender, but the sender cannot determine this by itself. This is why we introduce the echo mechanism; the EDS of a system transmits a control packet in the control-channel period that can be understood only by other stations of the same system, which can provide an echo; other systems perceive this as noise, while the EDS understands the signal.

Based on the channel organization assumptions described above, the proposed etiquette operates over a control channel organized into *frames*. The etiquette consists of framing mechanisms and mechanisms for the reservation of data channels. All this is done without having the system share a common PHY-layer protocol.

Framing is accomplished using the time of transmission as the only feedback to systems. A frame in the control channel consists of a framing signal followed by a sequence of channel-control periods. There is one channel-control period for each data channel, and there is a unique predefined association between a channel-control period and a data channel. The framing signal consists of a transmission pattern guaranteed to differ from any pattern within and across boundaries of channel-control periods.

The signaling used in each control period is based on the notion that different systems can only detect signal duration from one another (perceived as noise), while stations in the same system can actually exchange data. Thus, the activity on the control channel will be in the form of request-echo pairs. The EDS of a system will transmit a certain information packet in the control-channel period which will be understood only by other stations of the same system, who could provide and echo. Other systems will perceive this as noise. Of course, if more than a single EDS transmit concurrently a collision occurs and everybody perceives this as noise.

Data channels are reserved by means of requests made dynamically by the EDS. Because such requests are made when stations in a system require a data channel, requests from different systems may occur at the same time. Although such collisions could be resolved using a random backoff approach similar to what simple MAC protocols do (e.g., ALOHA, CSMA), an etiquette based on such an approach would not be stable and could not guarantee a maximum delay for a system to acquire a data channel. Because of the desirability of providing channel-assignment delay guarantees, we designed our etiquette using a deterministic tree-splitting collision resolution algorithm [2] tailored for the case in which the number of systems competing for the available data channels is finite.

Fig. 1 shows an example of an etiquette frame with three channel-control periods suitable for a band with three data channels. Let $\tau$ be the maximum propagation time for systems in the band, and let $\gamma$ be the duration of a reservation request (as well as the length of an echo signal). We define $\delta = \gamma + \tau$, which is the time required for a transmitted signal to be received. As was indicated earlier and as the Fig. 1 illustrates, the framing signal and channel-control periods consists of slots of duration $\delta$.

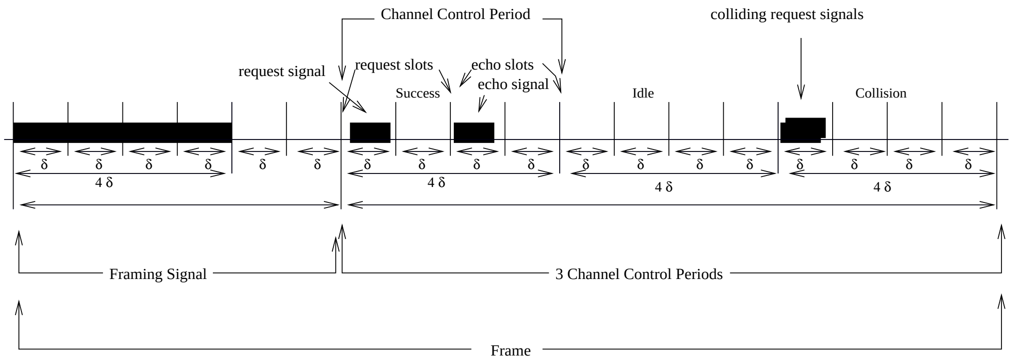


Figure 1: An example of an etiquette frame with three channel-control periods.

Because the collision resolution algorithm used to resolve channel requests can take multiple frames, the framing signal at the beginning of each frame must also specify whether or not an unfinished round of collision resolution is taking place in the present frame. This informs EDS with new channel requests to wait until the current round of requests is satisfied. In the proposed etiquette, a framing signal consisting of four consecutive slots with jamming (transmission of a signal by any of the systems), followed by two idle slots indicates the beginning of a *new* collision-resolution round; This is the case shown in Fig. 1. A framing signal consisting of three jamming slots followed by two idle slots indicates the *continuation* of a collision-resolution round.

Each channel-control period consists of four slots. These four slots are encoded to inform all systems of the availability, assignment, or ongoing contention for the corresponding data channel. The first two slots of the channel-control period are control slots used by systems to indicate current ownership of, or a request for the corresponding data channel. The last two slots of a channel-control period are echo slots used to provide feedback.

### 3.1 Initializing Frames

Let $s$ be the number of data channels defined on the band. If a system wishes to use one of the data channels its EDS first listens to the activity on the control channel. If the channel is idle for at least $4s\delta$ seconds (where $s$ is the number of data channels in the band) plus the duration of a framing signal, the EDS will send a framing signal of duration $4\delta$ followed by two idle slots ($2\delta$). This determines the beginning of a new reservation frame as well as a new collision-resolution round. Each EDS requesting the use of a channel sends a request signal of size $\gamma$ within the first slot of the four slots associated with the respective channel-control period. The request signal is followed by an idle period of size $\delta$. If the request was the only one, the intended receiver within the same system will understand the signal and send an echo signal back. For all the other systems in the network this request signal is noise, therefore, they will not send an echo signal.

We denote by $< abcd >$ the encoding used in any given channel-control period, where a,b,c and d are 0 or 1, depending on whether the corresponding slot is silent (idle) or there is a signal from at least one system. In terms of this notation, a station must listen for $< 111100 >$ to detect a new frame that does not have an ongoing round of request resolution. The signal $< 11100 >$ indicates that the frame has an ongoing resolution round and the station must refrain from requesting a channel.

Note that because any two EDSs are within $\tau$ seconds of one another, and because the encoding used in the channel-control periods prohibits four or three consecutive jamming slots, all active EDSs detect the beginning of the frame at the end of last jamming slot sent by any such station for framing purposes. Also note that a station must listen for an entire frame duration before attempting to request any channel. This implies that a station knows the state of the band assignment when it makes its request.

### 3.2 Signaling in the Channel-Control Period

Each station requesting the use of the channel sends a request signal of size $\gamma$ within the first slot of the four slots assigned to the respective channel-control period. The request signal is always followed by an idle period of size $\delta$. If only one system requested the channel, one or multiple receivers within the same system understand the signal and one of them send an echo signal back in the third slot of the channel-control period. For all the other systems, the request signal appears as noise for which they do not send an echo signal. This case corresponds, therefore, to an encoding of $< 1010 >$ in the channel-control period of the channel. If the request was unsuccessful due to a collision of multiple requests, no station will understand the request and consequently none will respond in the echo slot. This results in a channel control period of $< 1000 >$. A channel-control period with an empty signal $< 0000 >$ corresponds to an unused data channel.

We have seen that a successful request for a channel is encoded by $< 1010 >$, an unsuccessful request for the channel is $< 1000 >$ and an empty channel by $< 0000 >$. It is clear that, because a station in a system needs feedback within its own system, a period must include at least two slots, one for a request and one for the echo to the request. The reason why four slots are used to encode each channel-control period is that a system must also convey the following additional information:

- The code $< 0010 >$ is used to signal that the corresponding data channel is busy but was not the last data channel in the band to be reserved. This is important in the collision resolution algorithm as described later.
- The code $< 0001 >$ signals that the corresponding data channel is busy and it was the last data channel in the band to be reserved. Other EDSs will attempt to reserve data channels starting with the next data channel in the band. This code tells EDSs wishing to request a channel to start their bids on the next channel. As soon as a new last data channel is reserved the EDS changes the code from $< 0001 >$ to $< 0010 >$ in the next frame.

In addition, because of the need to guarantee uniqueness of the framing signal, no sequence of channel-control periods may contain a code of $< 1111 >$, $< 1110 >$, or $< 0111 >$ which are prefixes of framing signals. This is achieved by having the second slot of a channel-control period always be 0.

Once a system acquires a channel it can keep using it as long as it needs it or until it is challenged by another system. A system that has acquired a channel must send an echo signal in each subsequent frame to ensure the continuing use of such channel. If

no echo signal is sent other systems are free to make use of the data channel. If the current system is challenged by a new system, the new system sends a request signal. The old system notices the request signal, i.e., it is aware of a signal in the request slot, and refrains from emitting an echo signal. Since the new system understands the request signal and such signal was not interfered by the old system, an echo signal reserving the channel is transmitted. If two or more systems send a request signal within the same slot, none of the systems will understand it, therefore, no echo will be sent. Each system involved in the collision will select a new channel among the free channels. If all the channels are busy, the system must select randomly among one of the busy channel-control periods.

### 3.3 Resolving Channel Requests Conflicts

Because a single EDS in each system interacts with EDSs from other systems, we describe how channel requests are resolved by referring to a system making the request, rather than the EDSs.

In the proposed etiquette, a system that requires access to a data channel listens to the control channel for an entire frame to ascertain the state of the band, i.e., which data channels are free, whether there is an ongoing resolution of channel requests, and which was the last data channel to be assigned.

Each system is assigned a unique identifier, and maintains a stack, and two variables ($LowID$ and $HiID$). $LowID$ is initially the lowest ID and $HiID$ the highest ID given to any system. Together, they constitute the allowed ID interval that can attempt to reserve a channel. If the ID of a system is not within the allowed interval, it cannot request a channel. The stack is simply a storage mechanism for ID intervals that are waiting to get permission to request a channel.

Initially, all channels are free and there is no activity in the control channel or in the channels corresponding to each channel-control period.

When a passive system requires a channel, it first listens to the control channel. When the framing signal is detected, it listens for the entire frame, and records the state of each of the $s$ channels. If the control channel is idle (i.e., no framing signal is found) for a period equivalent to the size of a frame ($4s\delta+4\delta+2\delta$ seconds) the system transmits a framing signal. The framing signal determines the beginning of the frame.

When a system attempting to reserve a channel detects that there is an unfinished round of channel request resolution, which is detected when the framing signal is $< 11100 >$, the system waits until it reads an entire frame starting with a framing signal $< 111100 >$ indicating that the prior channel allocation requests have been resolved. Starting with the first channel-control period, all systems wishing to acquire a channel transmit a request signal in the first slot (the request slot) of the first channel-control period leaving the next slot idle. The sender then waits and listens to the channel for one slot for an echo signal. An echo signal is transmitted by one or multiple stations in the same system only if the request signal is heard free from errors. This is the case if the system is the only one that transmitted a request signal. If noise is detected in the request slot the echo slot is left idle. An empty echo slot is interpreted by the requesting systems as a collision of channel requests.

If an echo signal is received, the system acquires the channel and begins transmitting its data in the corresponding data channel. The system has unlimited use of the band, until it is challenged by another system or until it does not need the channel anymore, after which the system releases the channel by stopping the transmission of request and echo signals. As long as a system maintains access to a data channel, it transmits the code $< 0010 >$ in the corresponding control period of each frame if the data channel was not the last channel assigned during the last resolution round, and transmits the code $< 0001 >$ otherwise. This permits all systems that need access to a new data channel to begin their requests with the next unused data channel following the data channel with a control period having a code of $< 0001 >$. An unused data channel is one for which its control period had a code of $< 0000 >$ (empty) or $< 1000 >$ (a collision of two or more requests and no current system in the channel) in the previous frame.

If the sender of a request signal does not receive an echo during the echo slot, the sender and all other systems participating in the etiquette know that a collision of requests has occurred. As soon as the first collision takes place, every system divides the ID interval $(LowID, HiID)$ into two ID intervals. The first ID interval is $(LowID, LowID + \lceil \frac{HiID+LowID}{2} \rceil - 1)$, which we will call the backoff ID interval, while the second ID interval is $(LowID + \lceil \frac{HiID+LowID}{2} \rceil, HiID)$ and is called the allowed ID interval. Each system updates the stack by executing a PUSH stack command, where the key being pushed is the backoff ID interval. After this is done, the system updates $LowID$ and $HiID$ with the values from the allowed ID interval. This procedure is repeated each time a collision is detected.

Only those systems that were involved in the first collision are allowed into the collision-resolution phase. All other systems are in REMOTE state and simply keep track of the state for each channel, as well as the allowed ID interval and the backoff ID interval. A system remain in REMOTE state remains in this state until all collisions are resolved from the previous round.

Collision resolution of requests evolves in terms of collision-resolution intervals, of which there are three cases: idle (i.e., code $< 0000 >$), success (i.e., code $< 1010 >$), or collision interval (i.e., code $< 1000 >$). In the first interval of the collision-resolution phase all systems in the allowed ID interval that are in the REQUEST state try to retransmit a request signal. If none of the systems within this ID interval request the channel (i.e., code $< 0000 >$), a new update of the stack and of the variables $LowID$ and $HiID$ is due. Each system executes a POP command in the stack. This new ID interval now becomes the new $HiID$ and $LowID$. The same procedure takes place if, during the first collision-resolution interval, only one system is requesting the channel; the originator receives the echo signal (i.e., the code $< 1010 >$ occurs) and the system begins transmission in the assigned channel. The third case of a collision-resolution interval is for multiple systems to request the same channel, causing a collision (i.e., code $< 1000 >$). The systems in the allowed ID interval are once more split into two new ID intervals and the stack as well as the variables for each system is updated.

The etiquette repeats the above steps, until all the requests have been resolved. Notice that, as soon as the backoff stack becomes empty and there are no values in the allowed interval, all systems know that all the collisions of channel requests have been resolved for the of requests resolution and a new round can start, if there are systems that require data channels.

### 3.4 Example of the Etiquette's Operation

We illustrate the etiquette's operation using a simple example (see Fig. 2) with four systems labeled $n_{00}$, $n_{01}$, $n_{10}$, and $n_{11}$, and three channel-control periods per frame, labeled $s1$, $s2$ and $s3$. The framing signals (i.e., $< 111100 >$ and $< 11100 >$) are omitted for simplicity and we consider one round of collision resolution. We also assume that once a data channel is busy it remains busy until it is challenge by another system. Because the example assumes the beginning of a new request-resolution round, the backoff stack is empty and the allowed ID interval contains all the systems in the network, i.e., the allowed ID interval is $(n_{00}, n_{11})$ (Step 0 in Fig. 2).

At the end of the previous request-resolution round in the example, systems $n_{11}$ and $n_{10}$ have acquired channels $s3$ and $s1$ respectively (Step 0 in Fig. 2). Based on the state of the channels, all systems know that the channel $s1$ was the last data channel in the band to be reserved (code $< 0010 >$), while channel $s3$ was busy but was not the last data channel in the band to be reserved (code $< 0010 >$). If all channels are busy, the next contention channel is the next data channel after the channel with code $< 0001 >$. On

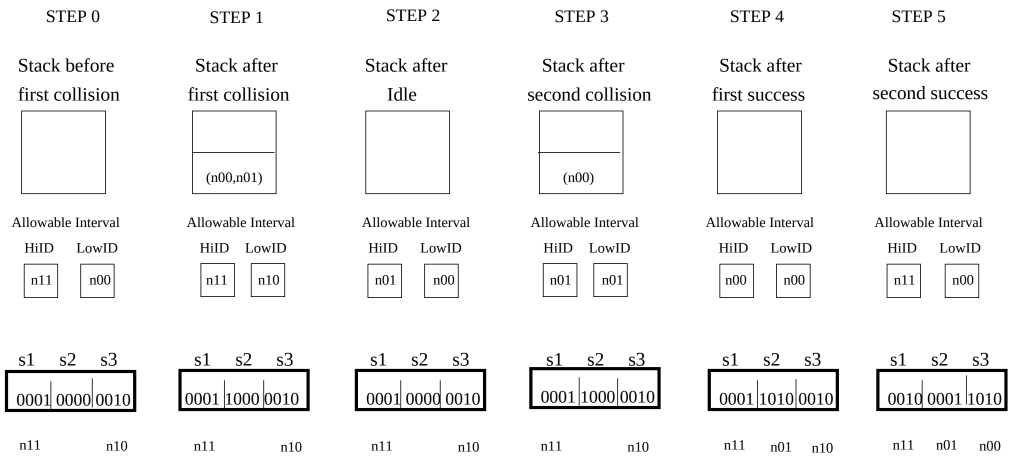


Figure 2: An etiquette's operation for four systems with three channel-control periods. The state of the last round is illustrated in step 0; systems $n_{11}$ and $n_{10}$ have acquired two out of the three available channels; systems $n_{00}$ and $n_{01}$ request a channel in the next round.

the other hand, if there are data channels that are not currently busy (code $< 0000 >$) or with an unsuccessful request (code $< 1000 >$), the next contention will be done in the next empty or unsuccessful request channel following the data channel with code $< 0001 >$. Therefore, the next contention in our example occurs in channel $s2$.

After the framing signal $< 111100 >$ is transmitted by all the active participating systems, all systems notice the beginning of a new request-resolution round. System $n_{11}$ continues sending the echo signal in $s1$ (code $< 0001 >$), indicating that the channel is still in use. Systems $n_{00}$ and $n_{01}$ use channel $s2$ to request a data channel since both are in the allowed ID interval. Although systems $n_{10}$ and $n_{11}$ are within the allowed interval they do not participate in the contention of channel $s2$ because they already have acquired a data channel. The first collision in channel $s2$ occurs (Step 1 in Fig. 2) with systems $n_{00}$ and $n_{01}$ each sending a request signal. If the request was unsuccessful due to a collision of multiple request, no feedback exists (i.e., corresponding to code $< 1000 >$); the backup stack and the allowed ID interval are updated. Systems $n_{00}$ and $n_{01}$ are members of the backoff ID interval; therefore, they both are on hold, they must wait until the collisions in the allowed ID interval are resolved. In the next frame, systems $n_{10}$ and $n_{11}$ are allowed to request data channel $s2$. Finally, the unsuccessful request in channel $s2$ is followed by an echo signal from system $n_{10}$ (code $< 0010 >$) in channel $s3$, terminating the first frame.

The second frame is initiated with the signal $< 11100 >$ transmitted by all the active participating systems. System $n_{11}$ continues sending the echo signal in the first channel-control period $s1$ (code $< 0001 >$). In the next channel-control period, $s2$, an idle period occurs (code $< 0000 >$, see Step 2 in Fig. 2), because systems $n_{10}$ and $n_{11}$ are in the allowed ID interval but do not need to request the channel. At the end of the channel-control period, all systems notice that the code was $< 0000 >$, which means that there were no collisions; accordingly, the systems in the system must update their intervals and the stack. They execute a POP-stack command and the new allowed interval is $(n_{00}, n_{01})$ (Step 2 in Fig. 2). The idle channel-control period for channel $s2$ is followed by an echo signal from system $n_{10}$ (code $< 0010 >$) in the $s3$ channel-control period, terminating the second frame.

The third frame is initiated with the framing signal $< 11100 >$ transmitted by all the active participating systems. System $n_{11}$ continues sending the echo signal in the first channel-control period $s1$ (code $< 0001 >$). In the second channel-control period both systems $n_{00}$ and $n_{01}$ transmit an echo signal (Step 3 in Fig. 2) and another collision occurs. Because a collision occurred, the allowed ID interval is split, i.e., system $n_{01}$ is within the allowed interval while the $n_{00}$ system must wait, its interval is the top of the stack. The third frame terminates with system $n_{10}$ sending the echo signal in the third channel-control period $s3$.

The fourth frame is initiated with the signal $< 11100 >$ transmitted by all the active participating systems. System $n_{11}$ continues sending the echo signal in the first channel-control period $s1$ (code $< 0001 >$). Since in the second channel-control period only one system is is in the allowed ID interval, system $n_{01}$ acquires channel $s2$ (code $< 1010 >$ in Step 4 in Fig. 2). At the end of the channel-control period the systems do an update, i.e., a POP-stack command. System $n_{00}$ is the new allowed ID interval and the backup stack is empty. The fourth frame terminates with system $n_{10}$ sending an echo signal in the third channel-control period $s3$. All three channels are busy, therefore, the next contention is done in the channel-control period following the code $< 1010 >$. In the example the contention is continued in $s3$ because $s2$ was the last busy data channel to be reserved.

The fifth frame is initiated with the framing signal $< 11100 >$. System $n_{11}$ continues sending the echo signal in the first channel-control period $s1$. Because $s1$ is no longer the last data channel to be reserved the code $< 0001 >$ is replaced by the code $< 0010 >$. In the second channel-control period system $n_{01}$ sends the echo signal $< 0001 >$ instead of code $< 1010 >$. In the third channel-control period System $n_{00}$ can request and acquire data channel (Step 5 in Fig. 2). At the end of the third channel-control period the systems do an update, i.e., a POP-stack command. Both the backup stack and the allowed ID interval are empty. The termination of the collision-resolution phase is determined by an empty stack and an empty allowed ID interval. The systems empty their stacks and update the allowed ID interval permitting all systems to contend in the next request-resolution round.

## 4 Etiquette Performance

In this section we show that the performance of the proposed etiquette approaches that of an optimal assignment of channels to systems from the standpoint of data channel utilization. We obtain a lower bound on the etiquette's throughput. We begin our analysis by finding the average number of steps required until $m$ channel request are resolved. A step is define as a channel control period (idle, success, collision) and has the length of $4\delta$. We then derive the throughput of any given system.

### 4.1 Average Number of Request-Resolution Steps

Let there be $n$ systems in the network, each with a distinct ID and $m \leq n$ of the systems request one data channel each. The total number of data channels available is $s$ and are assume empty, i.e., unused. All $m$ systems compete for the first channel and sequentially continue in the next data-control period as describe in the example in Section 3.4, until all the $m$ requests are resolved.

**Theorem 1** *Let there be $m > 1$ requests for channel assignment from $m$ distinct systems (one request per system) and let there be $n >= m$ maximum number total systems, then the average number of steps required until all $m$ channels requests are resolved is*

$$\overline{\mathcal{T}}(n,m) = \sum_{i=\mu}^{\nu} \frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-i) + \overline{\mathcal{T}}(\beta, i) + 1\right] \tag{1}$$

*where*

$$\begin{aligned}
\alpha &= \lceil n/2 \rceil;\ \beta = n - \alpha = n - \lceil n/2 \rceil \\
\mu &= \begin{cases} 0 & \text{if } m \leq \alpha \\ m - \alpha & \text{if } m > \alpha \end{cases} \\
\nu &= \begin{cases} m & \text{if } m \leq \beta \\ \beta & \text{if } m > \beta \end{cases}
\end{aligned}$$

**Proof:** We define a step as a channel control period of size $4\delta$. It is trivial that for all $n \geq 1$, $\overline{\mathcal{T}}(n,0)$ and $\overline{\mathcal{T}}(n,1)$ equal 1, i.e., we need in each of these cases one step. If we have in total two systems and both send a request signal within the same request slot, the average number of steps is $\overline{\mathcal{T}}(2,2) = 3$, one for the collision and two for the two successful request/echo exchanges.

With this initial conditions and following the tree-splitting algorithm we are in a position to find the average number of steps for $\overline{\mathcal{T}}(3,2)$. Since $m = 2$ we have to split the three total number of systems ($n = 3$) into two splits $\alpha = 2$ and $\beta = 1$ respectively. Therefore, we can either have 2 systems requesting a channel in the $\alpha$-split and none in the $\beta$-split; or 1 system in the $\alpha$-split and 1 in the $\beta$-split. The probability that 2 systems requesting a channel are in the $\alpha$-split while the remaining 0 requesting systems are in the $\beta$-split is given by $P\{(2 \in \alpha) \wedge (0 \in \beta)\} = \frac{\binom{2}{2}\binom{1}{0}}{\binom{3}{2}}$ and the probability that 1 systems requesting a channel is in the $\alpha$-split and 1 requesting systems is in the $\beta$-split is given by $P\{(1 \in \alpha) \wedge (0 \in \beta)\} = \frac{\binom{2}{1}\binom{1}{1}}{\binom{3}{2}}$. For each of these cases the probability of the split must be multiply by the average number of steps for the right split plus the average number of steps for the left split plus one step for the root of both splits. Therefore,

$$\overline{\mathcal{T}}(3,2) = \sum_{i=0}^{1} \frac{\binom{2}{2-i}\binom{1}{i}}{\binom{3}{2}} \left[\overline{\mathcal{T}}(2, 2-i) + \overline{\mathcal{T}}(1, i) + 1\right] \tag{2}$$

We assume that, for all $\alpha$ and $\beta$, the average number of steps $\overline{\mathcal{T}}(\alpha,m)$ and $\overline{\mathcal{T}}(\beta,m)$ are known. If $n$ is even, $\alpha = \beta = \frac{n}{2}$; otherwise, $\beta = \alpha - 1$. The probability that $m - i$ systems requesting a channel are in the $\alpha$-split while the remaining $i$ requesting systems are in the $\beta$-split is given by $P\{(m-i \in \alpha) \wedge (i \in \beta)\} = \frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}}$. Therefore, the average number of steps for this specific split, i.e., $m - i$ systems in the $\alpha$-split and $i$ systems in the $\beta$-split is equal to the average number of steps for the $\alpha$-split, plus the average number of steps for the $\beta$-split, plus one step for the root of both splits, times the probability of such a split, i.e., $\frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-i) + \overline{\mathcal{T}}(\beta, i) + 1\right]$.

For the average number of steps, $\mathcal{T}(n,m)$, we need to consider the cost as well as the probability of each of the possible $\alpha$- and $\beta$-splits. Therefore,

$$\begin{aligned}
\overline{\mathcal{T}}(n,m) &= \frac{\binom{\alpha}{m-\mu}\binom{\beta}{\mu}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-\mu) + \overline{\mathcal{T}}(\beta, \mu) + 1\right] + \ldots \\
&\ldots + \frac{\binom{\alpha}{m-\nu}\binom{\beta}{\nu}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-\nu) + \overline{\mathcal{T}}(\beta, \nu) + 1\right]
\end{aligned} \tag{3}$$

There are three possible $\mu$-$\nu$ combinations. First, if $m \leq \alpha$ and $m \leq \beta$, then $\mu = 0$ and $\nu = m$. In the second case, $m \leq \alpha$ and $\beta < m$; therefore, $\mu = 0$, while $\nu = \beta$. Finally, if $m$ is greater than both $\alpha$ and $\beta$, then $\mu = m - \alpha$ and $\nu = \beta$. Note that the parameter $m$ cannot be $> \alpha$ and $\leq \beta$ at the same time because $\beta \leq \alpha$; accordingly, this case is excluded. The sum of the average number of steps for each of the possible splits yields Eq (1). ∎

**Theorem 2** *Starting with $s$ empty channels and $m$ out of the $n$ total number systems requesting the use of a channel, the total number of channel-control periods required until the $k$th successful request/echo signal exchange is*

$$\begin{aligned}
\overline{\mathcal{T}}(n,m,k) = &\sum_{i=\mu}^{m-k} \frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-i, k) + 1\right] + \\
&\sum_{i=m-k+1}^{\nu} \frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-i, m-i) + \overline{\mathcal{T}}(\beta, i, i+k-m) + 1\right]
\end{aligned} \tag{4}$$

**Proof:** It can clearly be seen that if all $k$ successes are within the $\alpha$-split the steps in the $\beta$-split can be dropped altogether. Therefore, if we stop the recursion in Eq. (1) as soon as the $k$th successful request/echo signal exchange is achieved, than Eq. (1) can be rewritten as Eq. (4). ∎

If we set $k = 1$ in Eq. (4) we get the average number of steps up to the first successful request/echo exchange.

$$\begin{aligned}
\overline{\mathcal{T}}(n,m,1) = &\sum_{i=\mu}^{m-1} \frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-i, 1) + 1\right] + \\
&\sum_{i=m}^{\nu} \frac{\binom{\alpha}{m-i}\binom{\beta}{i}}{\binom{n}{m}} \left[\overline{\mathcal{T}}(\alpha, m-i, m-i) + \overline{\mathcal{T}}(\beta, i, i+1-m) + 1\right]
\end{aligned} \tag{5}$$

According to the etiquette, if all the $s$ channels are busy, each of the $m$ original systems requesting a channel contend in the same channel-control period. The collision-resolution steps are executed in each consecutive channel control period allocating channels as the resolution progresses; therefore, the number of channel control periods needed until all $m$ systems are allocated a channel is given by Eq. (1).

### 4.2 Etiquette's Throughput

We define the average throughput of a data channel in the band as

$$S = \frac{\bar{t}_{in}}{\bar{t}_{in} + \bar{t}_{out}} \tag{6}$$

where $\bar{t}_{in}$ is the average busy period for any given system, i.e., the amount of time during which the system is using the channel to

transmit data. $\bar{t}_{out}$ is the average acquisition delay, i.e., the average interval between two consecutive busy periods. $\bar{t}_{in}$ can also be visualized as the average duration a system spends in a channel before it is forced to release the channel, and $\bar{t}_{out}$ is the access delay or the average duration that it takes a system to acquire a channel.

We will first assume a network with $s$ data channels and $n$ total number of systems, out of which $m$ systems compete to acquire a channel. We assume that a system can at most acquire one channel at any given time. In the first part of the analysis we are interested in knowing the average number of frames required until all $s$ channels are being use if we have $m$ new systems trying to acquire a data channel for the case that we start with all $s$ channels free of users.

For all $n \geq m \geq k \geq 1$, Theorem 2 determines the average number of steps $\overline{\mathcal{T}}(n, m, k)$ required for up to $k$ successful request/echo exchanges, while Theorem 1 determines the average number of steps $\overline{\mathcal{T}}(n, m)$ required until all $m$ collisions are resolved. Therefore, because there are $s$ steps per frame, the average number of frames $\overline{\mathcal{F}}(n, m, s)$ required until all $m$ systems are assigned a channel is

$$\overline{\mathcal{F}}(n, m, s) = \lceil \overline{\mathcal{T}}(n, m)/s \rceil \quad \text{if} \quad m \leq s \tag{7}$$

We can compare $\overline{\mathcal{F}}(n, m, s)$ to the optimal case in which all $m$ systems are assigned $m$ channels in exactly $m$ steps. The optimal case assumes that there are only successful request/echo exchanges. Therefore, the total number of frames required for the optimal case is

$$\overline{\mathcal{F}}(n, m, s) = \lceil m/s \rceil \quad \text{if} \quad m \leq s \tag{8}$$

Fig. 3 shows the results for the analysis as well as the simulation. In the simulation, the total number of systems in the network ($n$) was set to 100 and the number of channels ($s$) was set to 30. Starting with $s$ empty channels $m$ random systems requested a data channel. For each $m$, 100 trials were simulated, each with $m$ different systems requesting a data channel. For each trial we kept track of: (a) the acquisition delay, i.e., the time (measured in frames) required for each requesting system to acquire a data channel; (b) the busy period, i.e., the time a given system uses the data channel before it releases the channel; and (c) the throughput, i.e., the ratio of time the given system is busy versus the total time.

As shown in Fig. 3, the proposed etiquette requires twice the number of frames compared to the optimal etiquette. The optimal etiquette would allocate the request in a strictly linear number of steps. It is a theoretical lower bound and represents the best possible performance.

In the second part of the analysis we are interested in finding bounds on the delays for data-channel acquisition by any given system. We assume $m + s$ systems competing for $s$ channels at any given time. Systems that loose their channel wait until the end of the current collision-resolution round and try again in the next collision-resolution round. As soon as the collision-resolution round is over all the systems that lost their channels compete for a spot. We assume that in every collision-resolution round all the $s$ channels are being used and that $m$ systems compete to acquire a channel, i.e., in each collision-resolution round $m$ new systems enter replacing $m$ old systems. The old systems contend for a channel in the next collision resolution round according to the etiquette rules.

Let $x$ denote the number of full collision-resolution rounds from the time a system acquires a channel until it looses the channel, and $k$ an integer from 1 to $m$ denoting the request/echo exchange in which the given system acquires the channel. There can be at most $m$ request/echo exchanges per collision-resolution round, and $k'$ denotes the number of successful request/echo exchanges in the last collision-resolution round before a given system looses the channel. Let us also define $\bar{t}_{in}$ as the average time any given system uses a channel before it must release it, i.e., the average busy period. The throughput of any given system is obtained directly from the following two theorems.

**Theorem 3** *For $m + s$ systems in a network the average busy period for any given system given that $m \leq s$ is*

$$\bar{t}_{in} = \frac{1}{m} \sum_{k=1}^{m} (1 + x)\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, k) + \overline{\mathcal{T}}(n, m, k') \tag{9}$$

**Proof:** Assume that all the $s$ channels are busy and a given system acquires a channel at the $k$th successful request/echo exchange within a collision-resolution round of length $\overline{\mathcal{T}}(n, m)$ steps. If $m \leq s$, $s$ successful request/echo exchanges must take place before the system entering at $k$th success must give up the reserved channel. This is true since the collision-resolution algorithm persists in the same channel until a success is achieve moving to the next channel-control period. Therefore, at the end of the first collision-resolution round $\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, k)$ frames later we have hat $m - k$ new systems acquiring a channel, i.e., we still have $s - m + k$ systems requesting a channel before the $k$th system has to release the channel. Therefore, there are $x = \lceil (s + k - m - 1)/m \rceil$ collision-resolution rounds in between the first round (when the given system acquired a channel) and the last collision-resolution round (when the given system looses the channel). $k'$ is deterministic and is a function of $k$ and $s$. It can be express as $k' = s + k - (1 + x)m$. Therefore, for a given $k$, $t_{in}$ can be written as

$$t_{in} = (\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, k)) + x\overline{\mathcal{T}}(n, m) + \overline{\mathcal{T}}(n, m, k') \tag{10}$$

The value for $\bar{t}_{in}$ can be found by averaging over all the possible $k$ values which is given by Eq. (9). ∎

Eq. (9) is bounded by

$$\begin{aligned} (1 + x)\overline{\mathcal{T}}(n, m) &- \overline{\mathcal{T}}(n, m, m) + \overline{\mathcal{T}}(n, m, 1) \leq \overline{t_{in}} \\ &\leq (1 + x)\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, 1) + \overline{\mathcal{T}}(n, m, m) \end{aligned} \tag{11}$$

In Fig. 6 the average busy period $t_{in}$ measure in frames is plotted.

**Theorem 4** *For $m + s$ systems the average interval between two busy periods for any given system given that $m \leq s$ is bounded by*

$$\overline{\mathcal{T}}(n, m, 1) + 1 \leq \overline{t_{out}} \leq 2\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, 1) + 1 \tag{12}$$

*where $t_{out}^{UB} = 2\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, 1) + 1$ s the upper bound for the average interval between busy periods.*

**Proof:** Assume that after the $k$th successful request/echo exchange within a collision-resolution round the given system looses the channel. The given system must wait until the end of the collision-resolution round before it can make a request, i.e., it must wait $\overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, k)$ frames. In the next collision-resolution round the given system will acquire a channel, where $k'$ can range from 1 to $m$. An extra frame must be added since once a successful request/echo exchange has taken the given system must wait until the end of the frame before sending its data using the channel. Therefore,

$$t_{out} = \overline{\mathcal{T}}(n, m) - \overline{\mathcal{T}}(n, m, k) + \overline{\mathcal{T}}(n, m, k') + 1 \tag{13}$$

The lower bound can be found by setting $k = m$ and $k' = 1$. Respectively, the upper bound can be found if $k = 1$ and $k' = m$

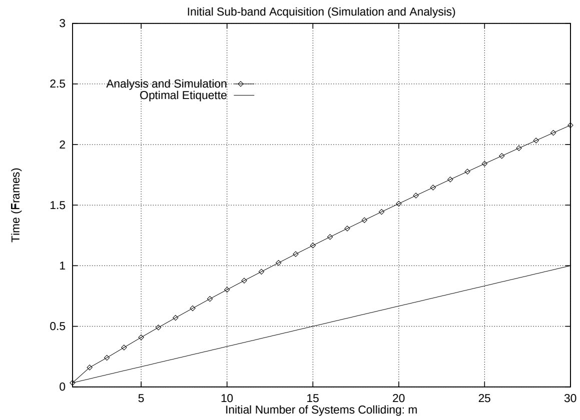


Figure 3: Total number of frames needed to resolve $m$ initial collisions.

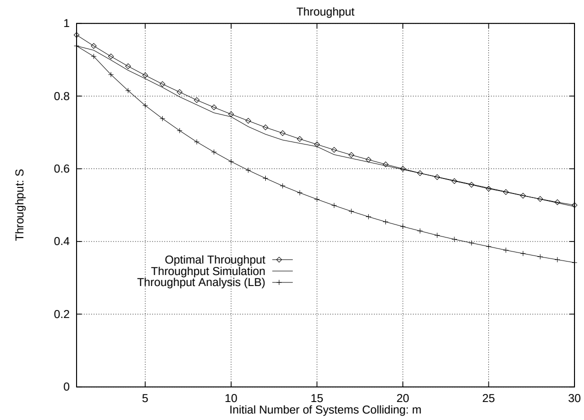


Figure 4: Throughput for the optimal etiquette, i.e. the upper bound, simulation, and the lower bound as a function of $m$ initial collisions.

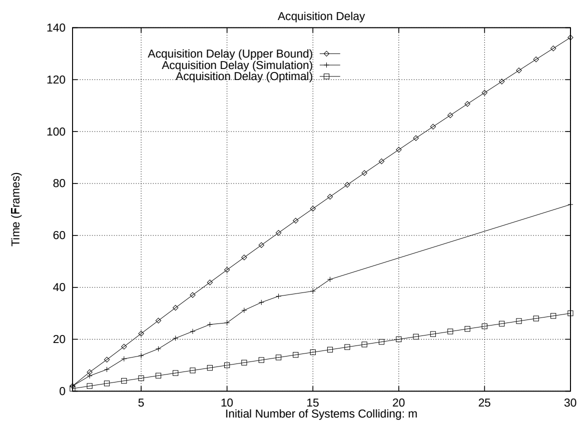


Figure 5: Total acquisition delay measured in frames as a function of $m$ initial collisions.

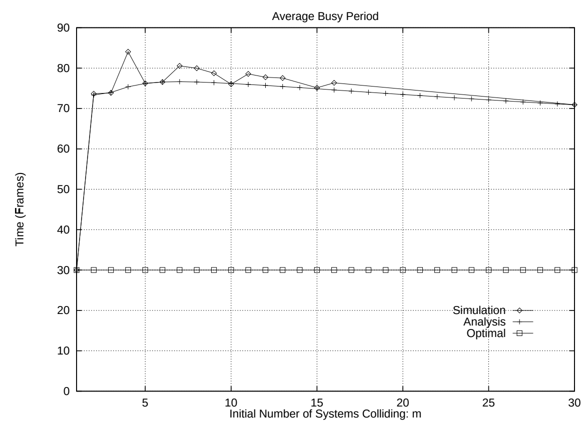


Figure 6: Average busy period measured in frames as a function of $m$ initial collisions.

are set in the above equation. Therefore, $t_{out}$ is bounded according to Eq. (12). ∎

In Fig. 5 the acquisition delay $t_{out}$ measure in frames is plotted. Given $t_{in}$ in Theorem 3, $t_{out}^{UB}$ in Theorem 4 for $m + s$ systems in a network and $m \leq s$, the throughput for any given system is bounded by

$$\mathcal{S} \leq \frac{\bar{t}_{in}}{\bar{t}_{in} + t_{out}^{LB}} \tag{14}$$

Fig. 4 shows simulation results and the bounds for the throughput.

## 5 Conclusion

We have proposed a specific set of access rules ("Spectrum Etiquette") for the general $59 - 59.05$ GHz band. The proposed etiquette permits heterogeneous systems to co-exist with one another by means of transmissions over a control channel used to establish collision-free transmission schedules over the channels allocated for data transmission within the 59-64 GHz band. The etiquette consists of framing and signaling rules that allow systems with different PHY protocol layers to communicate, and a request resolution algorithm that assigns data channels to systems with a performance that is closed to optimum under any load of channel assignment request.

## 6 References


[1] D. Bertsekas and R. Gallager, *Data Networks*, Second Edition, Prentice-Hall, 1992.

[2] J.I. Capetanakis, "Tree algorithm for packet broadcasting channel," *IEEE Trans. Inform. Theory*, vol.IT-25, pp. 505-515, Sept. 1979.

[3] L. Dellaverson and W. Dellaverson, "Distributed Channel Access on Wireless ATM Links," *IEEE Communications Magazine*, November, 1997.

[4] Federal Communications Commission, "First Report and Order and Second Notice of Proposal Rulemaking," *http://www.fcc.gov/oet/dockets/et94-124*, December 15, 1995.

[5] M. J. Karol, Liu Zhao, and K. Y. Eng, "Distributed-queuing request update multiple access (DQRUMA) for wireless packet (ATM) networks," *Proc. IEEE ICC '95*, Seattle, WA, USA, June 18-21, 1995.

[6] The Millimeter Wave Communications Working Group, "Report and Recommendations of the MWCWG to the FCC," *http://www-uk.hpl.hp.com/projects/funfair/public/default.html*, December 13, 1996.

[7] I. Vukovic and J. McKown, "Spectrum sharing under the asynchronous UPCS etiquette: The performance of collocated systems under heavy load," *Proc. ACM/IEEE Mobile Computing and Networking '96*, Rye, NY, Nov. 10-12, 1996.

[8] W. Xu and G. Campbell, "A distributed queuing random access protocol for a broadcast channel," *Proc. ACM SIGCOMM'93*, San Francisco, USA, 13–17 Sept. 1993.